PATENT ABSTRACTS OF JAPAN

(11)Publication number : **2001-092869**

(43)Date of publication of application : **06.04.2001**

(51)Int.Cl. **G06F 17/50**
H01L 21/82

(21)Application number : **11-269802**

(22)Date of filing : **24.09.1999**

(71)Applicant : **NEC ENG LTD**

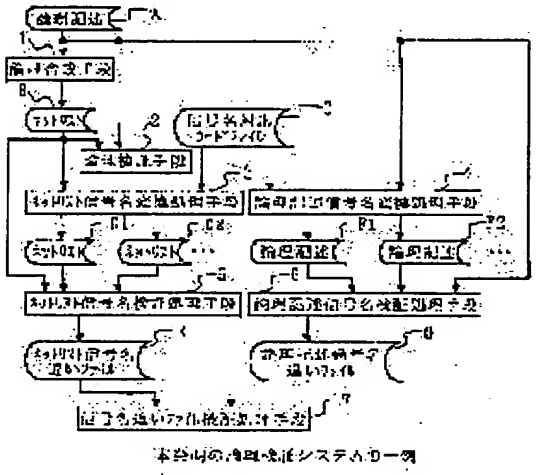
(72)Inventor : **HASHIDA TOUSHI**

(54) DEVICE AND METHOD FOR SYNTHESIZING AND VERIFYING LOGIC AND RECORDING MEDIUM

(57)Abstract:

PROBLEM TO BE SOLVED: To reduce the man-hour required for synthesizing and verifying a plurality of net lists from logical descriptions.

SOLUTION: A means 1 generates a net list B from a logical description A and a means 2 verifies the consistency between the description A and list B. Means 3 and 4 generate net lists DI, etc., corresponding to different signal names, and logical descriptions F1, etc., by converting the signal names in the list B and description A in accordance with a signal name corresponding card file C in which different signal names are arranged correspondingly to the signal names in the list B. A means 5 detects the difference in signal name between the net list B and net lists DI, etc., and outputs a net list signal name difference file E in which the signal names are arranged. A means 6 detects the difference in signal name between the logical description A and logical descriptions F1, etc., and outputs a logical description signal name difference file G in which the signal names are arranged. A means 7 verifies the consistency between the files E and G.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of

rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] About the logic synthesis and the verification system in the LSI design which uses two or more general-purpose logic macroes (netlist), especially, this invention creates two or more netlists from one kind of logic description, and relates to the logic verification system which verifies the consistency of logic description and each netlist.

[0002]

[Description of the Prior Art] In the design of an integrated circuit, logic synthesis generates the netlist which expressed the connection relation of a circuit in the unit of the circuit element of gate level as everyone knows from the text description (it is hereafter described as "logic description".) which described logical circuit information with the hardware description language (HDL). By LSI, although signals differ, two or more logical circuits which perform the same logical operation exist in many cases, and two or more netlists with which signal names differ from one kind of logic description are especially generated in that case. Therefore, the logic verification which verifies the consistency of logic description and each netlist becomes an important activity.

[0003] The example of the conventional logic synthesis and verification system which performs such logic synthesis and verification is shown in drawing 14 . the logic description a which serves as a radical in drawing 14 at the beginning -- two or more sheets -- copying -- two or more logic description a1 and a2 ... is obtained. next, each logic description a1 and a2 -- it changes into the signal name which changes with signal name conversion means 30 to ... two or more logic description a11 and a21 from which a signal name differs to the logic description a used as a radical by this although logic is the same ... can be created. Next, the logic description a which became a radical and logic description a11 and a21 which conversion of a signal name finished ... To each, the logic synthesis means 10 performs logic synthesis, and they are netlists b, b1, and b2... It creates. the last -- the logic verification means 20 -- each logic description a, a11, and a21 ..., netlist b.b1, and b2 -- consistency with ... is verified about each.

[0004] That is, this logic synthesis and verification system create two or more logic

description and two or more netlists from one kind of logic description, and verifies multiple times about the consistency of the logic description and the netlist which correspond, respectively.

[0005]

[Problem(s) to be Solved by the Invention] However, in the conventional logic synthesis and verification system mentioned above, since it is necessary to do a logic synthesis activity two or more times in order to create two or more logic description from which it consists of same logic and a signal name differs and to create each netlist to each logic description, there is the 1st trouble that the manday of a logic synthesis activity increases.

[0006] Moreover, since it is necessary to do a logic verification activity two or more times in order to perform logic verification between the netlists corresponding to two or more logic description and each logic description, there is the 2nd trouble that the manday of a logic verification activity increases.

[0007] Furthermore, since two or more logic description from which it consists of same logic and a signal name differs exists and correction of two or more logic description languages is needed when modification of logic arises, there is the 3rd trouble that possibility that the correction mistake by buildup of correction manday and the break in of a help activity will arise becomes high.

[0008] The object of this invention is to offer the logic synthesis system which mitigated the manday of the logic synthesis activity which creates two or more netlists from one kind of logic description.

[0009] Moreover, other objects of this invention are to offer the logic verification system which mitigated the manday of a logic verification activity which verifies the consistency between one kind of logic description, and two or more netlists.

[0010]

[Means for Solving the Problem] In the logic synthesizer unit with which the logic synthesizer unit of the 1st this invention creates two or more netlists with which signal names differ from one kind of logic description By changing the signal name in said netlist into a logic synthesis means to create a netlist, and the signal name response in this netlist from said logic description according to the card file corresponding to the signal name which enumerated said different signal names It is characterized by having with a netlist signal name transform-

processing means to create said netlist corresponding to a different signal name. [0011] In the logic verification equipment which the logic synthesizer unit of the 2nd this invention creates two or more netlists with which signal names differ from one kind of logic description, and performs logic verification of the this created netlist A logic synthesis means to create a netlist from said logic description, and a logic verification means to verify the consistency between said logic description and these netlists, By changing the signal name in said netlist into the signal name response in this netlist according to the card file corresponding to the signal name which enumerated said different signal names A netlist signal name transform-processing means to create said netlist corresponding to a different signal name, By changing the signal name under said logic description into the signal name response in this netlist according to the card file corresponding to the signal name which enumerated said different signal names A logic description signal name transform-processing means to create said logic description corresponding to a different signal name, The difference in the signal name between the netlist created from said logic description and the netlist after said signal name conversion is detected. A netlist signal name verification means to output the netlist signal name difference file which enumerated the signal name, The difference in the signal name between said logic description and the logic description after said signal name conversion is detected. It is characterized by having a logic description signal name verification means to output the logic description signal name difference file which enumerated the signal name, and a signal name difference file verification means to verify the consistency between said netlist signal name difference file and said logic description signal name difference file.

[0012] In the logic synthesis approach that the logic synthesis approach of the 1st this invention creates two or more netlists with which signal names differ from one kind of logic description By changing the signal name in said netlist into the logic synthesis procedure which creates a netlist, and the signal name response in this netlist from said logic description according to the card file corresponding to the signal name which enumerated said different signal names It is characterized by having the netlist signal name transform-processing procedure which creates said netlist corresponding to a different signal name.

[0013] In the logic verification approach of the logic synthesis approach of the 2nd this invention creating two or more netlists with which signal names differ from one kind of logic

description, and performing logic verification of the this created netlist The logic synthesis procedure which creates a netlist from said logic description, and the logic verification procedure of verifying the consistency between said logic description and these netlists, By changing the signal name in said netlist into the signal name response in this netlist according to the card file corresponding to the signal name which enumerated said different signal names The netlist signal name transform-processing procedure which creates said netlist corresponding to a different signal name, By changing the signal name under said logic description into the signal name response in this netlist according to the card file corresponding to the signal name which enumerated said different signal names The logic description signal name transform-processing procedure which creates said logic description corresponding to a different signal name, The difference in the signal name between the netlist created from said logic description and the netlist after said signal name conversion is detected. The netlist signal name verification procedure which outputs the netlist signal name difference file which enumerated the signal name, The difference in the signal name between said logic description and the logic description after said signal name conversion is detected. It is characterized by having the logic description signal name verification procedure which outputs the logic description signal name difference file which enumerated the signal name, and the signal name difference file verification procedure of verifying the consistency between said netlist signal name difference file and said logic description signal name difference file.

[0014]

[Embodiment of the Invention] Next, the gestalt of operation of this invention is explained to a detail with reference to a drawing.

[0015] Reference of drawing 1 shows the logic synthesis and the verification system as a gestalt of 1 operation of this invention. as for the gestalt of this operation, the logic synthesis means 1 does a logic synthesis activity only once to one kind of logic description A which becomes a radical -- one netlist B -- creating -- the netlist signal name transform-processing means 3 -- the signal name of this netlist B -- changing -- two or more netlists D1 and D2 ... is created. moreover, the logic description signal name transform-processing means 4 -- the signal name of the logic description A -- changing -- two or more logic description F1 and F2 ... is created.

[0016] The logic verification means 2 also performs logic verification only once between the logic description A and Netlist B. And for the rest, the netlist signal name verification processing means 5 is Netlist B and netlists D1 and D2... The netlist signal name difference of a between is detected. Moreover, the logic description signal niece verification processing means 6 is the logic description A and the logic description F1 and F2... The logic description signal name difference of a between is detected, and, finally the signal name difference file verification processing means 7 verifies the consistency between a netlist signal name difference and a logic description signal name difference.

[0017] First, each I / O file is explained. The logic description A is the text description which described logical circuit information by HDL, and is an input file used as the input of the logic synthesis means 1. Netlist B is the file created by the logic synthesis means 1, and is a list which expressed the connection relation of a logical circuit in the unit of the circuit element of gate level. The card file C corresponding to a signal name is a file which enumerated the responses of the signal name described by the logic description A inputted into the logic synthesis means 1, and the signal name after conversion. netlists D1 and D2 -- it is the file which ... changed the signal of Netlist B by the card file C corresponding to a signal name, and the netlist signal name transform-processing means 5 created. the logic description F1 and F2 -- it is the file to which the logic description signal name transform-processing means 4 created ... by the card file C corresponding to a signal name. The netlist signal name difference file E is Netlist B and the netlists D1 and D2 which the netlist signal name verification processing means 5 created... It is the file which enumerated the differences in the signal name of a between. And the logic description signal name difference file G is the logic description A and the logic description F1 and F2 which the logic description signal name verification processing means 6 created... It is the file which enumerated the differences in the signal name of a between.

[0018] Next, the outline of the function of each means is explained. The logic synthesis means 1 inputs the logic description A, and outputs Netlist B. A logic verification means inputs Netlist B and the logic description A, and performs logic verification about the consistency between both. The logic synthesis means 1 and the logic verification means 2 do not have the conventional technique and the changing place.

[0019] two or more netlists D1 and D2 which the netlist signal name transform-processing

means 3 inputted the card file C and Netlist B corresponding to a signal name, and changed the signal name ... is outputted. two or more logic description F1 and F2 which the logic synthesis processing signal name transform-processing means 4 inputted the card file C and the logic description A corresponding to a signal name, and changed the signal name ... is outputted. the netlists D1 and D2 of the plurality [means / 5 / netlist signal name verification processing] after the netlist B after logic synthesis, and netlist signal name transform processing ... is inputted and the netlist signal name difference file E which enumerated different signal names among both is outputted. the logic description A and the logic description F1 and F2 of plurality [means / 6 / logic description signal name verification processing] after logic description signal name transform processing ... is inputted and the logic description signal name difference file G which enumerated different signal names among both is outputted. The signal name difference file verification processing means 7 inputs the netlist signal name difference file E and the logic description language signal name difference file G, and a signal name difference file is verified by detecting the difference in the signal name enumerated among both.

[0020] Next, actuation of the netlist signal name transform-processing means 3 is explained to a detail using the flow chart of drawing 2 .

[0021] The netlist in drawing 2 is the netlist B outputted from the logic synthesis means 1, and a card file points out the card file C corresponding to a signal. The processing which reads the line of the beginning of the netlist with which the logic synthesis means 1 outputted this netlist signal name transform processing (step S1 of drawing 2), The processing which judges whether the signal name defined as the line of the signal name described in the line of the netlist processed and (step S2) read which reads the line of the beginning of a card file, and the read card file is the same (step S3), The processing which judges whether the line read from the card file is a last line of a card file (step S4), The processing which changes the signal name of the line of the read netlist into the signal name of a card file, and is outputted to a new netlist (step S5), The processing which outputs the line of the netlist processed and (step S6) read which reads the next line of a card file to a new netlist (step S7), It consists of processings (step S9) which read the next line of the processing (step S8) which judges whether the line read from the netlist is a last line of a netlist, and a netlist.

[0022] Next, the procedure of processing is explained. Processing (step S1) which reads the

line of the beginning of a netlist first is performed. In the processing (step S1) which reads the line of the beginning of a netlist, the line (the 1st line) of the beginning of the netlist B used as the input to the netlist signal name transform-processing means 3 is read. And it moves to the processing (step S2) which reads the line of the beginning of a card file. In the processing (step S2) which reads the line of the beginning of a card file, the line (the 1st line) of the beginning of the card file C used as the input to the netlist signal name transform-processing means 3 is read. And it moves to the processing (step S3) which judges whether the signal name defined as the line of the signal name described in the line of the read netlist and the read card file is the same.

[0023] In the processing (step S3) which judges whether the signal name defined as the line of the signal name described in the line of the read netlist and the read card file is the same When the signal name defined as the line of the signal name described in the line of the read netlist B and the read card file C is the same It moves to the processing (step S5) which changes the signal name of the line of the read netlist B into the signal name of a card file C, and is outputted to a new netlist. When the signal name defined as the line of the signal name described in the line of the read netlist B on the other hand and the read card file C is not the same, it moves to the processing (step S4) which judges whether the line read from the card file is a last line of a card file.

[0024] In the processing (step S4) which judges whether the line read from the card file is a last line of a card file, it moves to the processing (step S7) which outputs the line of the read netlist to a new netlist when the line read from the card file C is a last line of a card file C, and on the other hand, when the line read from the card file C is not a last line of a card file C, it moves to the processing (step S6) which reads the next line of a card file.

[0025] the signal name corresponding to the signal name which defines the signal name description part of the line of the read netlist B as the line read by that of a card file C in the processing (step S5) which changes the signal name of the line of the read netlist into the signal name of a card file, and is outputted to a new netlist -- changing -- the new netlists D1 and D2 -- it outputs to ... And it moves to the processing (step S8) which judges whether the line read from the netlist is a last line of a netlist.

[0026] In the processing (step S6) which reads the next line of a card file, the next line of the card file C read now is read. And it moves to the processing (step S3) which judges whether

the signal name defined as the line of the signal name described in the line of the read netlist and the read card file is the same.

[0027] the line of Netlist C which read the line of the read netlist in the processing (step S7) outputted to a new netlist -- conversion of a signal name -- not carrying out -- the new netlists D1 and D2 -- it outputs to ... And it moves to the processing (step S8) which judges whether the line read from the netlist is a last line of a netlist.

[0028] In the processing (step S8) which judges whether the line read from the netlist is a last line of a netlist, when the line read from Netlist B is a last line of Netlist B, this processing is ended, and on the other hand, when the line read from Netlist B is not a last line of Netlist B, it moves to the processing (step S9) which reads the next line of a netlist.

[0029] In the processing (step S9) which reads the next line of a netlist, the next line of the netlist B read now is read. And it moves to the processing (step S2) which reads the line of the beginning of a card file.

[0030] the new netlists D1 and D2 which changed the signal name of Netlist B by the card file C corresponding to a signal name the above result ... is obtained. In addition, new netlists D1 and D2 ... A number is based on the number of the signal names matched with the signal name of Netlist B in the card file C corresponding to a signal name.

[0031] Next, actuation of the logic description signal name conversion means 4 is explained to a detail using the flow chart of drawing 3 . It is what was read as "logic description" about that there is drawing 3 with a "netlist" in drawing 2 , and the flow itself is the same.

[0032] The logic description in drawing 3 is the logic description A used as the input of the logic synthesis means 1, and a card file points out the card file C corresponding to a signal. The processing which reads the line of the beginning of the logic description from which this logic description signal name transform processing serves as an input of the logic synthesis means 1 (step S10 of drawing 3), The processing which judges whether the signal name defined as the line of the signal name which reads the line of the beginning of a card file, and which was processed (step S11) and has been described in the line of the read logic description, and the read card file is the same (step S12), The processing which judges whether the line read from the card file is a last line of a card file (step S13), Processing (step S14) the signal name of the line of the read logic description language -- the signal name of a card file -- changing -- the new logic description F1 and F2 -- it outputs to ... Processing (step

S16) the line of the logic description processed and (step S15) read which reads the next line of a card file -- the new logic description F1 and F2 -- it outputs to ... The line read from logic description consists of processings (step S18) which read the next line of the processing (step S17) which judges whether it is the last line of logic description, and logic description.

[0033] Next, the procedure of processing is explained. Processing (step S10) which reads the line of the beginning of logic description is performed to the beginning. In the processing (step S10) which reads the line of the beginning of logic description, the line (the 1st line) of the beginning of the logic description A used as the input to the logic description signal name conversion means 4 is read. And it moves to the processing (step S11) which reads the line of the beginning of a card file. In the processing (step S11) which reads the line of the beginning of a card file, the line (the 1st line) of the beginning of the card file C used as the input to the logic descriptive name transform-processing means 4 is read. And it moves to the processing (step S12) which judges whether the signal name defined as the line of the signal name described in the line of the read logic description language and the read card file is the same.

[0034] In the processing (step S12) which judges whether the signal name defined as the line of the signal name described in the line of the read logic description and the read card file is the same When the signal name defined as the line of the signal name described in the line of the read logic description A and the read card file C is the same It moves to the processing (step S14) which changes the signal name of the line of the read logic description A into the signal name of a card file C, and is outputted to a new logic description language. When the signal name defined as the line of the signal name described in the line of the read logic description A on the other hand and the read card file is not the same, it moves to the processing (step S13) which judges whether the line read from the card file is a last line of a card file.

[0035] It moves to the processing (step S16) which outputs the line of the read logic description to new logic description when the line read from the card file C is a last line of a card file C, and when the line read from the card file C is not a last line of a card file C, on the other hand in the processing (step S13) which judges whether the line read from the card file is a last line of a card file, it moves to the processing (step S15) which reads the next line of a card file.

[0036] the signal name corresponding to the signal name which defines the signal name

description part of the line of the read logic description A as the line which the card file C read in the processing (step S14) which changes the signal name of the line of the read logic description into the signal name of a card file, and is outputted to new logic description -- changing -- new logic description F1 and F2 -- it outputs to ... And the line read from logic description moves to the processing (step S17) which judges whether it is the last line of logic description.

[0037] In the processing (step S15) which reads the next line of a card file, the next line of the card file C read now is read. And it moves to the processing (step S12) which judges whether the signal name defined as the line of the signal name described in the line of the read logic description and the read card file is the same.

[0038] the line of the logic description A which read the line of the read logic description in the processing (step S16) outputted to new logic description -- conversion of a signal name -- not carrying out -- the new logic description F1 and F2 -- it outputs to ... And the line read from logic description moves to the processing (step S17) which judges whether it is the last line of logic description.

[0039] In the processing (step S17) which judges whether the line read from logic description is a last line of logic description, when the line read from the logic description A is a last line of the logic description A, this processing is ended, and on the other hand, when the line read from the logic description A is not a last line of the logic description A, it moves to the processing (step S18) which reads the next line of a logic description language.

[0040] In the processing (step S18) which reads the next line of logic description, the next line of the logic description A read now is read. And it moves to the processing (step S11) which reads the line of the beginning of a card file.

[0041] the new logic description F1 and F2 which changed the signal name of the logic description A by the card file C corresponding to a signal name the above result ... is obtained. In addition, new logic description F1 and F2 ... A number is based on the number of the signal names matched with the signal of the logic description A in the card file C corresponding to a signal name.

[0042] Next, actuation of the netlist signal name verification processing means 5 is explained to a detail using the flow chart of drawing 4.

[0043] the netlist B outputted from the logic synthesis means 1 with all the netlists in drawing

4 , and the netlists D1 and D2 outputted from the network signal name transform-processing means 3 ... is said. This netlist signal name verification processing consists of the processing (step S19) which reads the line of the beginning of all netlists, or or the processing (step S20) which makes a judgment with the same signal name of all netlists, processing (step S21) which enumerate and output a signal name, processing (step S22) which judge whether the line read from the netlist is a last line, and processing (step S23) which read the next line of TTORISUTO altogether.

[0044] Next, the procedure of processing is explained. First, processing (step S19) which reads the line of the beginning of all netlists is performed. In the processing (step S19) which reads the line of the beginning of all netlists, the first line (the 1st line) is read from all netlists. And it moves to the processing (step S20) which judges whether the signal name described in the line read from all netlists is the same. In the processing (step S20) which judges whether the signal name described in the line read from all netlists is the same, it judges whether the signal name described in the line read from all netlists is the same. And when a signal name is the same, it moves to the processing (step S22) which judges whether the line read from all netlists is a last line, and when a signal name is not the same, on the other hand, it moves to the processing (step S21) which enumerates and outputs a signal name.

[0045] In the processing (step S21) which enumerates and outputs a signal name, the signal name description part of the line read from all netlists is enumerated and outputted. And it moves to the processing (step S22) which judges whether the line read from all netlists is a last line of a netlist.

[0046] In the processing (step S22) which judges whether the line read from all netlists is a last line, it judges whether the line read from all netlists is a last line. When it is a last line, this processing is ended, and on the other hand, when it is not a last line, it moves to the processing (step S23) which reads the next line of all netlists.

[0047] In the processing (step S23) which reads the next line of all netlists, the next line of all the netlists read now is read. And it moves to the processing (step S20) which judges whether the signal name described in the line read from all netlists is the same.

[0048] Next, actuation of the logic description signal name verification processing means 6 is explained to a detail using the flow chart of drawing 5 . It is what was read as "all logic

description" about that there is drawing 5 with "all netlists" in drawing 4 , and the flow itself is the same.

[0049] the logic description A which becomes the input of the logic synthesis means 1 with all logic description in drawing 5 , and the logic description F1 and F2 outputted from the logic description signal name transform-processing means 6 ... is said. The processing whose the logic description signal name verification processing of this reads the line of the beginning of all logic description (step S24), The processing which judges whether the signal name of all logic description is the same (step S25), It consists of processings (step S28) which read the next line of the processing (step S26) which enumerates and outputs a signal name, the processing (step S27) which judges whether the line read from all logic description is a last line, and all logic description.

[0050] Next, the means of processing is explained. First, processing (step S24) which reads the line of the beginning of all logic description is performed. In the processing (step S24) which reads the line of the beginning of all logic description, the first line (the 1st line) is read from all logic description. And it moves to the processing (step S25) which judges whether the signal name described in the line read from all logic description is the same. In the processing (step S25) which judges whether the signal name described in the line read from all logic description is the same, it judges whether the signal name described in the line read from all logic description is the same. And when a signal name is the same, it moves to the processing (step S27) which judges whether the line read from all logic description is a last line, and on the other hand, when a signal name is not the same, it moves to the processing (step S26) which enumerates and outputs a signal name.

[0051] In the processing (step S26) which enumerates and outputs a signal name, a signal name description part is enumerated and outputted from all logic description. And it moves to the processing (step S28) which judges whether the line read from all logic description is a last line.

[0052] In the processing (step S27) which judges whether the line read from all logic description is a last line, it judges whether the line read from all logic description is a last line. When it is a last line, this processing is ended, and on the other hand, when it is not a last line, it moves to the processing (step S28) which reads the next line of all logic description.

[0053] In the processing (step S28) which reads the next line of all logic description, the next

line of all logic description read now is read. And it moves to the processing (step S25) which judges whether the signal name described in the line read from all logic description is the same.

[0054] Next, actuation of the netlist signal name difference file verification processing means 7 is explained to a detail using the flow chart of drawing 6 .

[0055] The processing whose the signal name ** file verification processing of this reads the line of the beginning of a netlist signal name difference file (step S29), The processing which reads the line of the beginning of a logic description signal name difference file (step S30), The content with the read same line, the processing (step S31) to judge, the processing which the line read from the logic description signal name difference file judges in a last line (step S32), The processing (step S33) which outputs an error message, the processing which reads the next line of a logic description signal name difference file (step S34), The line read from the netlist signal name difference file consists of processings (step S36) which read the next line of a last line, the processing (step S35) to judge, and a netlist signal name difference file.

[0056] Next, the procedure of processing is explained. First, processing (step S29) which reads the line of the beginning of a netlist signal name difference file is performed. In the processing (step S29) which reads the line of the beginning of a netlist signal name file, the line (the 1st line) of the beginning of the netlist signal name difference file E is read. And it moves to the processing (step S30) which reads the line of the beginning of a logic description signal name file. In the processing (step S30) which reads the line of the beginning of a logic description signal name difference file, the line which read and read moves from the line (the 1st line) of the beginning of the logic description signal name difference file G to the same content or the processing (step S31) which judges.

[0057] In the content with the read same line, or the processing (step S31) to judge, the read line judges in the same content. When the read line is the same, the line read from the netlist signal name difference file E moves to a last line or the processing (step S35) to judge, and on the other hand, when the read line is not the same, the line read from the logic description signal name difference file G moves to a last line or the processing (step S32) to judge.

[0058] When the line which the line read from the logic description signal name difference file read from the logic description signal name difference file G in a last line or the processing (step S32) to judge is a last line, it moves to the processing (step S33) which

outputs an error message, and when it is not a last line, it moves to the processing (step S34) which reads the next line of a logic description language signal name file.

[0059] In the processing (step S33) which outputs an error message, an error is outputted and the line read from the netlist signal name file moves to a last line or the processing (step S35) to judge. In the processing (step S34) which reads the next line of a logic description signal name difference file, it moves from a logic description signal name file difference to the processing (step S31) reading and the read line judge the following line to be in the same content.

[0060] When the line which the line read from the netlist signal name difference file read from the netlist signal name difference file E in a last line or the processing (step S35) to judge is a last line, this processing is ended, and when it is not a last line, it moves to the processing (step S36) which reads the next line of a netlist signal name difference file.

[0061] In the processing (step S36) which reads the next line of a netlist signal name difference file, it moves from a netlist signal name file difference to the processing (step S30) which reads the line of the beginning of a reading and logic description signal name file for the following line.

[0062] Next, an example is shown and the logic synthesis and verification by this invention are explained. Drawing 7 shows the example of the logic description A about the signal name A which described logical circuit information of acquiring a deed of the AND operation of a signal AA 1 and a signal AA 2, and a signal AA 3. To this logic description A, the netlist B after the logic synthesis by the logic synthesis means 1 is as drawing 8 , and logic is expressing the circuit connection relation that a signal AA 3 is connected, to the signal AA 1 and the input terminal 2 at the signal A2 and the output terminal at AND and an input terminal 1. Moreover, the example of the card file C corresponding to the signal name used for this example is shown in drawing 9 . In drawing 9 , three signal names, BBB, CCC, and DDD, are enumerated as a signal name corresponding to the signal name AAA. Therefore, by the signal name BBB, each of signals DD1, DD2, and DD3 corresponds to signals AA1, AA2, and AA3 at signals BB1, BB2, and BB3 and the signal name CCC by signals CC1, CC2, and CC3 and the signal name DDD.

[0063] In the netlist signal transformation processing means 3, the line "AAA;BBB;CCC;DDD;" of the beginning of the card file C (drawing 9) corresponding to a

reading (step S1) and signal name is first read for the line "DID AAA" of the beginning of Netlist B (drawing 8) as shown in drawing 2 (step S2). Since the signal name of Netlist B is compared with the signal name of the card file C corresponding to a signal name (step S3), consequently the signal name AAA is defined as both sides and it is in agreement The netlist D1 (drawing 10 (1)) which changed the signal name AAA into the signal name BBB, The netlist D3 (drawing 10 (3)) which changed into the signal name DDD the netlist D2 (drawing 10 (2)) which changed the signal name AAA into the signal name CCC, and the signal name AAA is outputted (step S5). And since Netlist B still is not an end (step S8), the next line "RONRI AND" of Netlist B is read (step S9).

[0064] Again the line "AAA;BBB;CCC;DDD;" of the beginning of the card file C corresponding to a signal name Reading (step S2), Not shortly in agreement the signal name of Netlist B -- comparing (step S3) -- Moreover, since it is not the end of the card file C corresponding to a signal name (step S4), the next line "AA1;BB1;CC1;DD1;" of the card file C corresponding to a signal name is compared with reading (step S6) and the signal name of Netlist B (step S3). However, although read to the line "AA3;BB3;CC;DD3;" of the last of the card file C corresponding to a signal name, since it is not in agreement with the signal name of the 2nd line of Netlist B, the signal name of the 2nd line "RONRI AND" of Netlist B is outputted to netlists D1, D2, and D3 as it is (step S7).

[0065] Next, the line "AAA;BBB;CCC;DDD;" of the beginning of the card file C corresponding to a reading (step S9) and signal name is read for the 3rd ("INPUT1;AA1;) line of Netlist B (step S2). Since the signal name "AA1" defined as Netlist B is not defined as the line of the card file C corresponding to the signal name currently read now (step S3), to the card file C corresponding to a signal name, there is a definition or it reads the card file C corresponding to a signal name in order to the line of the last of the card file C corresponding to a signal name (step S3, S4). The signal name "AA1" defined as Netlist B Since it is defined as the 2nd line "AA1;BB1;CC1;DD1;" of the card file C corresponding to a signal name The signal name "AA1" of Netlist B is changed into "BB1, CC1, DD1", respectively, and "INPUT;BB1;", "INPUT;CC1", and "INPUT;DD1" are outputted to netlists D1, D2, and D3 (step S5).

[0066] When it carries out also with the line after the 4th line of Netlist B until it results the above processing in the line of the last of Netlist B along with the flow chart of drawing 2

(step S8), netlists D1, D2, and D3 come to be shown in drawing 10 . Drawing 10 is known by that it is the difference of only the signal names AAA, BBB, CCC, and DDD as compared with drawing 8 .

[0067] About the logic description A, the logic description signal name transform-processing means 4 performs signal name transform processing (drawing 2) by the above-mentioned netlist signal name transform-processing means 3, and same transform processing, as shown in drawing 3 . Consequently, the logic description F3 (drawing 11 (3)) which changed into the signal name DDD the signal name AAA of the logic description F1 (drawing 11 (1)) which changed the signal name AAA of the logic description A into the signal name BBB, the logic description F2 (drawing 11 (2)) which changed the signal name AAA of the logic description A into the signal name CCC, and the logic description A can be obtained.

[0068] Next, about all the netlists B, D1, D2, and D3, i.e., netlists, the netlist signal name verification means 5 outputs the netlist signal name difference file E (drawing 12) which enumerated different signal names, as shown in drawing 4 . In drawing 4 The 1st line "DID AAA" of the line B of the beginning of all netlists, i.e., a netlist, the 1st line "DID BBB" of a netlist D1, the 1st line of a netlist D2 "DID CCC", Reading (step S19), consequently a signal name are those (step S20) which is not the same about the 1st line "DID DDD" of a netlist D3, and signal names are enumerated with "AAA;BBB;CCC;DDD;" to the netlist signal name difference file E (step S21).

[0069] since a netlist is not an end (step S22) -- the next line of all netlists -- reading (step S23) -- since it is defined as "RONRI;AND;" by all the netlists B, D1, D2, and D3, nothing is done but it shifts to the following step (step S20).

[0070] If the same processing as the above is performed along with the flowchart of drawing 4 about the 3rd line of all netlists or below, the netlist signal name difference file E as shown in drawing 12 can be obtained.

[0071] About all logic description A, i.e., logic description, the logic description F1, the logic description F2, and the logic description F3, the logic description signal name verification means 6 performs signal name verification processing (drawing 4) by the above-mentioned netlist signal name verification means 5, and same processing, as shown in drawing 5 . Consequently, the logic description signal name difference file G (drawing 13) which enumerated different signal names is outputted. Although it is the same so that clearly if

drawing 13 is contrasted with drawing 12 , this is a case with a normal verification result.
[0072] The signal name difference file verification processing means 7 completes logic verification processing by comparing the content of the netlist signal name difference file E (drawing 12) and the logic description signal name difference file G (drawing 13), as shown in drawing 6 . That is, it reads both files one line at a time (steps S29 and S30), and an error message will be outputted, if reading (steps S32 and S34) and the still read content are not in agreement to the line of the last of (step S31) logic description signal name difference file G when the read content is different (step S33). About a netlist and logic description, since this changed the signal name by the same card file corresponding to a signal name as mentioned above, if transform processing of a signal name is performed normally, as shown in drawing 12 and drawing 13 , it will make it basis for both sides to be in agreement.
[0073] In addition, you may make it make it perform by memorizing the program for performing the logic synthesis approach and the logic verification approach which were mentioned above by computer to the record medium which can computer read semiconductor memory, a magnetic disk, etc., and making it read into a computer. Such a program acts so that a computer may be made to demonstrate the same function as the logic synthesis means 1 shown in drawing 1 , the logic verification means 2, the netlist signal name transform-processing means 3, the logic description signal name conversion means 4, the netlist signal name verification processing means 5, the logic description signal name verification processing means 6, and the signal name difference file verification processing means 7.
[0074]
[Effect of the Invention] From one kind of logic description, I hear that the 1st effectiveness of this invention can mitigate the manday of the logic synthesis activity which creates two or more netlists, and there is. It is because it became possible for the reason to create one logic description language, and to do a logic synthesis activity once, and for the rest to consider the card file corresponding to a signal name as an input, and to create two or more netlists by signal name transform processing.
[0075] Moreover, I hear that the 2nd effectiveness of this invention can mitigate the manday of a logic verification activity which verifies the consistency between one kind of logic description, and two or more netlists, and there is. The reason is that logic verification which verifies the netlist which is to the bases created from the logic description which is to one kind

of bases, and the consistency between basic logic description is performed only once, the rest detects the difference between basic logic description, the ****** description which changed the signal name about each basic netlist, and the signal name of a netlist, and it verified the consistency between both.

[0076] Furthermore, when modification of logic arises, two above-mentioned effectiveness mitigates the activity manday of creation of a netlist, and logic verification, and brings about the repercussion effect that the correction mistakes by the break in of a help activity are reducible.

[Translation done.]

CLAIMS

[Claim(s)]

[Claim 1] In the logic synthesizer unit which creates two or more netlists with which signal names differ from one kind of logic description By changing the signal name in said netlist into a logic synthesis means to create a netlist, and the signal name response in this netlist from said logic description according to the card file corresponding to the signal name which enumerated said different signal names The logic synthesizer unit characterized by having with a netlist signal name transform-processing means to create said netlist corresponding to a different signal name.

[Claim 2] In the logic verification equipment which creates two or more netlists with which signal names differ from one kind of logic description, and performs logic verification of the this created netlist A logic synthesis means to create a netlist from said logic description, and a logic verification means to verify the consistency between said logic description and these netlists, By changing the signal name in said netlist into the signal name response in this netlist according to the card file corresponding to the signal name which enumerated said different signal names A netlist signal name transform-processing means to create said netlist corresponding to a different signal name, By changing the signal name under said logic description into the signal name response in this netlist according to the card file

corresponding to the signal name which enumerated said different signal names A logic description signal name transform-processing means to create said logic description corresponding to a different signal name, The difference in the signal name between the netlist created from said logic description and the netlist after said signal name conversion is detected. A netlist signal name verification means to output the netlist signal name difference file which enumerated the signal name, The difference in the signal name between said logic description and the logic description after said signal name conversion is detected. Logic verification equipment characterized by having a logic description signal name verification means to output the logic description signal name difference file which enumerated the signal name, and a signal name difference file verification means to verify the consistency between said netlist signal name difference file and said logic description signal name difference file.

[Claim 3] Logic verification equipment according to claim 2 characterized by outputting an error message in said signal name difference file verification means when the signal name in said netlist signal name difference file does not exist in said logic description signal name difference file.

[Claim 4] In the logic synthesis approach which creates two or more netlists with which signal names differ from one kind of logic description By changing the signal name in said netlist into the logic synthesis procedure which creates a netlist, and the signal name response in this netlist from said logic description according to the card file corresponding to the signal name which enumerated said different signal names The logic synthesis approach characterized by having the netlist signal name transform-processing procedure which creates said netlist corresponding to a different signal name.

[Claim 5] In the logic verification approach of creating two or more netlists with which signal names differ from one kind of logic description, and performing logic verification of the this created netlist The logic synthesis procedure which creates a netlist from said logic description, and the logic verification procedure of verifying the consistency between said logic description and these netlists, By changing the signal name in said netlist into the signal name response in this netlist according to the card file corresponding to the signal name which enumerated said different signal names The netlist signal name transform-processing procedure which creates said netlist corresponding to a different signal name, By changing the signal name under said logic description into the signal name response in this netlist

according to the card file corresponding to the signal name which enumerated said different signal names The logic description signal name transform-processing procedure which creates said logic description corresponding to a different signal name, The difference in the signal name between the netlist created from said logic description and the netlist after said signal name conversion is detected. The netlist signal name verification procedure which outputs the netlist signal name difference file which enumerated the signal name, The difference in the signal name between said logic description and the logic description after said signal name conversion is detected. The logic verification approach characterized by having the logic description signal name verification procedure which outputs the logic description signal name difference file which enumerated the signal name, and the signal name difference file verification procedure of verifying the consistency between said netlist signal name difference file and said logic description signal name difference file.

[Claim 6] The logic verification approach according to claim 5 characterized by outputting an error message in said signal name difference file verification procedure when the signal name in said netlist signal name difference file does not exist in said logic description signal name file.

difference

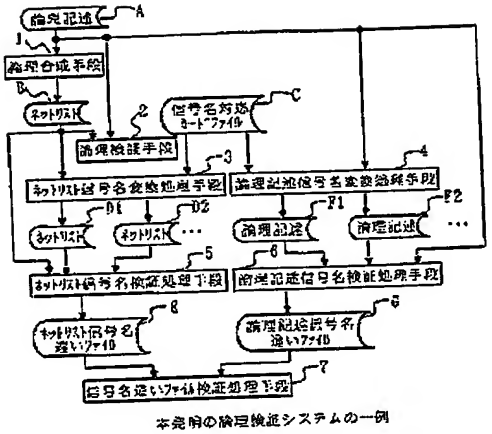
[Claim 7] The logic synthesis procedure which is the logic synthesis approach which creates two or more netlists with which signal names differ from one kind of logic description, and creates a netlist from said logic description, By changing the signal name in said netlist into the signal name response in this netlist according to the card file corresponding to the signal name which enumerated said different signal names The record medium which memorized the program for making a computer perform the logic synthesis approach characterized by having the netlist signal name transform-processing procedure which creates said netlist corresponding to a different signal name and in which computer reading is possible.

[Claim 8] The logic synthesis procedure which creates two or more netlists with which signal names differ from one kind of logic description, is the logic verification approach of performing logic verification of the this created netlist, and creates a netlist from said logic description, The logic verification procedure of verifying the consistency between said logic description and these netlists, By changing the signal name in said netlist into the signal name response in this netlist according to the card file corresponding to the signal name which enumerated said different signal names The netlist signal name transform-processing

procedure which creates said netlist corresponding to a different signal name, By changing the signal name under said logic description into the signal name response in this netlist according to the card file corresponding to the signal name which enumerated said different signal names The logic description signal name transform-processing procedure which creates said logic description corresponding to a different signal name, The difference in the signal name between the netlist created from said logic description and the netlist after said signal name conversion is detected. The netlist signal name verification procedure which outputs the netlist signal name difference file which enumerated the signal name, The difference in the signal name between said logic description and the logic description after said signal name conversion is detected. The logic description signal name verification procedure which outputs the logic description signal name difference file which enumerated the signal name, Said netlist signal name difference file The record medium which memorized the program for making a computer perform the logic verification approach characterized by having the signal name difference file verification procedure of verifying the consistency between said logic description signal name difference files and in which computer reading is possible.

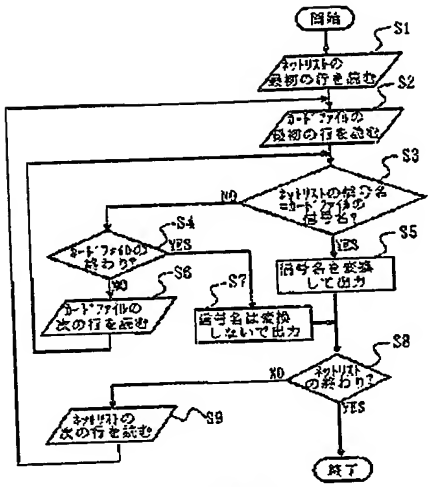
[Translation done.]

【図1】



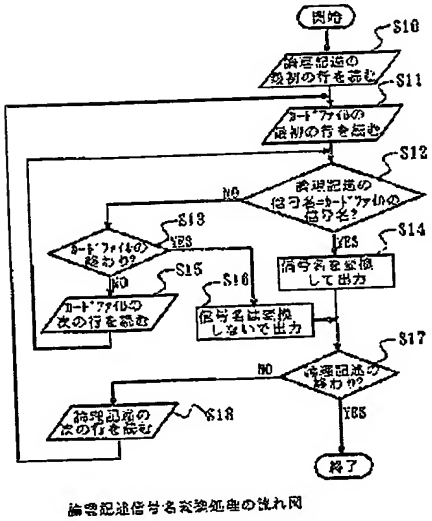
本発明の論理検証システムの一例

【図2】



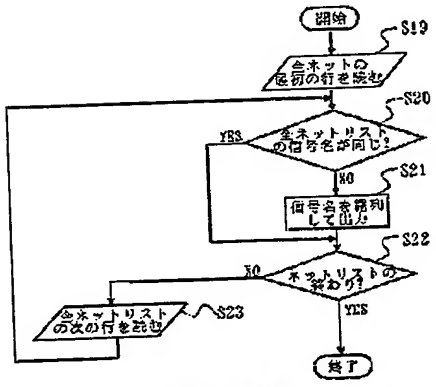
ネットリスト信号名変換処理の流れ図

【図3】



論理記述信号名変換処理の流れ図

【図4】



ネットリスト信号名違い検出処理の流れ図

【図12】

AAA;	BBB;	CCC;	DDD;
AA1;	BB1;	CC1;	DD1;
AA2;	BB2;	CC2;	DD2;
AA3;	BB3;	CC3;	DD3;

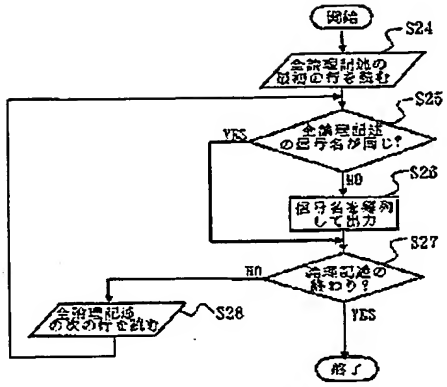
ネットリスト信号名違いファイルとの具体例

【図13】

AAA;	BBB;	CCC;	DDD;
AA1;	BB1;	CC1;	DD1;
AA2;	BB2;	CC2;	DD2;
AA3;	BB3;	CC3;	DD3;

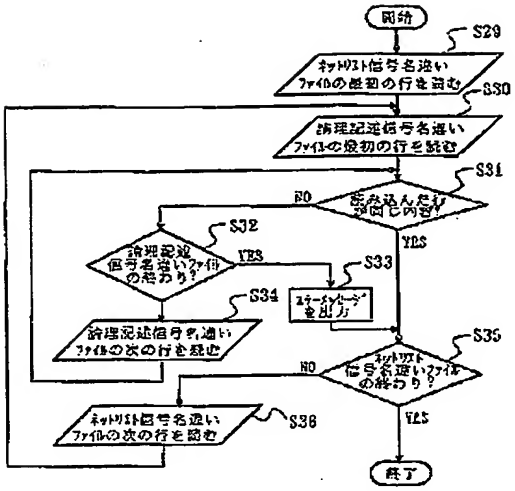
論理記述信号名違いファイルQの具体例

【図５】



論理記述信号名違い検出処理の流れ図

【図６】



【図１０】

```
DID;BBB;
RONRI;AND;
INPUT1;BB1;
INPUT2;BB2;
OUTPUT;BB3;
```

(1) 信号変換後のネットリストＤ１の具体例

```
DID;CCC;
RONRI;AND;
INPUT1;CC1;
INPUT2;CC2;
OUTPUT;CC3;
```

(2) 信号変換後のネットリストＤ２の具体例

```
DID;DDD;
RONRI;AND;
INPUT1;DD1;
INPUT2;DD2;
OUTPUT;DD3;
```

(3) 信号変換後のネットリストＤ３の具体例

【図１１】

```
DDD BBB
INPUT BB1、BB2
OUTPUT BB3
BB3=BB1・BB2
```

(1) 信号変換後の論理記述Ｆ１の具体例

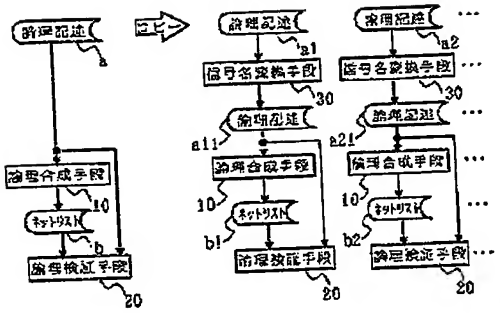
```
DDD CCC
INPUT CC1、CC2
OUTPUT CC3
CC3=CC1・CC2
```

(2) 信号変換後の論理記述Ｆ２の具体例

```
DDD DDD
INPUT DD1、DD2
OUTPUT DD3
DD3=DD1・DD2
```

(3) 信号変換後の論理記述Ｆ３の具体例

【図１４】



従来の論理検証システムの例